

Ispit iz predmeta Operativni sistemi 1

Ime i prezime: _____

Broj indeksa: _____

Broj poena: _____/30

Ispit traje 90 minuta. Nije dozvoljeno korišćenje literature.

1.(3) Šta je *afinitet* (*afinity*) procesa u simetričnom multiprocesorskom operativnom sistemu?

2.(3) Na assembleru picoRISC procesora napisati prevod sledeće rekurzivne funkcije:

```
int f (int n) { return (n<=0) ? 0 : f(n-1)+n; }
```

3.(3) Šta je TLB, čemu služi, tj. koji problem rešava i kako to postiže?

4.(3) Za neke vrste logičkih segmenata operativni sistem ne mora da učitava stranice koje im pripadaju iz *exe* fajla prilikom prvog adresiranja tih stranica. Navesti za koje sve vrste segmenata to važi i zašto njihove stranice ne moraju da se učitavaju.

5.(3) Ukoliko su svi sistemski pozivi izvršeni uspešno, koliko procesa se ukupno pokrene kada se nad sledećim programom pokrene jedan proces (računajući i taj jedan)?

```
void main () {  
    for (int i=0; i<5; i++) if (fork()>0) return;  
}
```

Odgovor: _____

6.(3) Tokom obrade nekih sistemskog poziva, za potrebe pozivajućeg procesa potrebno je više puta pokrenuti neku ulazno-izlaznu operaciju sa diskom, recimo učitavanje bloka od čijeg sadržaja zavisi dalji tok obrade tog sistemskog poziva u njegovom kodu ugnježenih potprograma. Tokom čekanja na završetak tih operacija treba suspendovati tu obradu i nastaviti je kad se operacija završi. Da li se obrada sistemskih poziva procesa u tom jezgru može odvijati tako da koristi isti, zajednički stek kernela? Obrazložiti.

7.(3) Korišćenjem funkcije *yield* koja prebacuje kontekst sa jednog na drugu datu nit, u školskom jezgru realizovati operaciju *wait* na semaforu.

8.(3) Navesti osnovne operacije klase blokovski orijentisanih uređaja sa direktnim pristupom.

9.(3) Neki fajl sistem pruža sledeće operacije u svom API za tekstualne fajlove:

- `int size(FHANDLE)` Vraća trenutnu veličinu sadržaja fajla u znakovima.
- `void append(FHANDLE,int)` Proširuje sadržaj fajla za dati broj znakova na kraju.
- `void seek(FHANDLE,int)` Postavlja kurzor datog fajla na datu poziciju (redni broj znaka počev od 0).
- `void write(FHANDLE,char*,int size)` Na poziciju kurzora datog fajla upisuje dati niz znakova zadate dužine, i pomera kurzor iza upisanog niza znakova.

Operacije `seek` i `write` rade samo u opsegu trenutne veličine sadržaja fajla (ne pomeraju kurzor i ne upisuju iza kraja sadržaja fajla). Realizovati funkciju

```
write(FHANDLE,int position,char*,int size);
```

koja na zadatu poziciju upisuje zadati niz znakova date veličine, pri čemu se fajl implicitno najpre proširuje na potrebnu veličinu ukoliko bi zadata pozicija ili zadati upis prekoračio trenutnu veličinu sadržaja fajla. Zanimariti sve moguće greške u ulazu/izlazu.

10.(3) Koliko pristupa blokovima na disku treba izvršiti za pristup n -tom logičkom bloku sadržaja fajla ako je alokacija fajla a) ulančana lista blokova, pri čemu na prvi blok sadržaja fajla ukazuje polje u FCB, b) indeksna, pri čemu je indeks fajla uvek u dva nivoa, a na blok sa indeksom prvog nivoa ukazuje polje u FCB? FCB fajla je u memoriji. Odgovor: a) _____ b) _____